

(11)特許出願公開番号

特開平11-340983

(43)公開日 平成11年(1999)12月10日

(51) Int.Cl.⁶
H 0 4 L 12/28

識別記号

F I
H O 4 L 11/20

G

審査請求 未請求 請求項の数13 OL (全 17 頁)

(21)出願番号 特願平11-130483

(22)出願日 平成11年(1999)5月11日

(31)優先權主張番号 9810089:4

(32)優先日 1998年5月11日

(33)優先権主張国 イギリス (GB)

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72)発明者 フィンバー ネイブン

イギリス国, チェシャー エスケー 8 5
ピーエフ, シードル ハルム, クロスフィ
ールド ロード 88

(72)発明者 ボール パーンズ

イギリス国, ダービーシャー エスケ-22
3 ピーアール, ハイ ピーク, ニュー
ミルズ, ホール ストリート 19

(74)代理人 弁理士 石田 敬 (外3名)

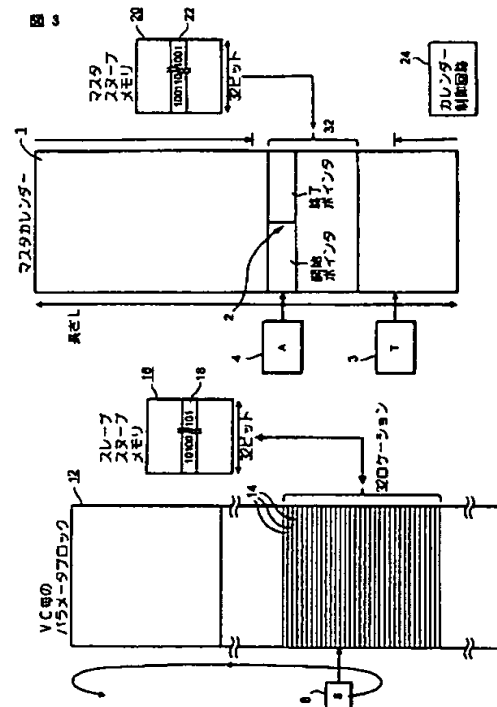
最終頁に続く

(54) 【発明の名称】 スケジューリング回路及び方法

(57) 【要約】

【課題】 スケジューリング回路に関し、特にATMネットワークユニット内で効率のよいセル転送をスケジュール可能なスケジューリング回路を提供する。

【解決手段】 マスタカレンダー１は所定のマスタカレンダースケジューリング範囲（ＳＲ）内で発生するイベントにそれぞれ対応したエントリを保持する。スレーブカレンダー１２は前記スケジューリング範囲外で発生するイベントにそれぞれ対応したエントリを保持する。イベントがスケジュールされる時、カレンダー制御回路２４は現時間とそのイベントの所望のスケジューリング時間との間隔が前記スケジューリング範囲を超える場合に、前記スレーブカレンダー１２にそれに対応したエントリを作成する。カレンダー制御回路２４は、さらに前記スレーブカレンダー１２のエントリをモニタし、前記エントリに対応するイベントが前記スケジューリング範囲内にあるエントリをマスタカレンダー１に転送する。



【特許請求の範囲】

【請求項1】 所定のイベントをスケジュールするためのスケジュールリング回路は、

所定のマスタカレンダースケジュールリング範囲内で発生するイベントにそれぞれ対応したエントリを保持するマスタカレンダー手段と、

前記スケジュールリング範囲外で発生するイベントにそれぞれ対応したエントリを保持するスレーブカレンダー手段と、

イベントがスケジュールされる時、現時間とそのイベントの所望のスケジュールリング時間との間隔が前記スケジュールリング範囲を超える場合に、前記スレーブカレンダー手段にそれに対応した前記所望のスケジュールリング時間を示すタイミング情報を含むエントリを作成するカレンダー制御手段とを有し、

前記カレンダー制御手段は、さらに前記スレーブカレンダー手段のエントリをモニタし、前記エントリに対応するイベントが前記スケジュールリング範囲内にあるエントリを前記マスタカレンダー手段に転送することを特徴とするスケジュールリング回路。

【請求項2】 前記カレンダー制御手段は、前記イベントのスケジュールリング中に、前記間隔がスケジュールリング範囲を超えない場合には、前記マスタカレンダー手段においてスケジュールされるイベントに対応したイベントを作成する請求項1記載のスケジュールリング回路。

【請求項3】 前記スレーブカレンダー手段の各エントリに含まれるタイミング情報の細分性は前記現時間を測定した細分性に等しく、

前記カレンダー制御手段は、前記スレーブカレンダー手段におけるエントリのモニタの間に、そのエントリに含まれるタイミング情報によって示される所望のスケジュールリング時間と前記現時間とを比較し、その比較結果により前記転送を実行するか否かを判断する請求項1又は2に記載のスケジュールリング回路。

【請求項4】 前記転送は、前記エントリのタイミング情報によって示される所望のスケジュールリング時間の差が、前記スケジュールリング範囲及び所定のオフセット時間の合計より小さいか等しい量だけ前記現時間と異なる時に実行される請求項3記載のスケジュールリング回路。

【請求項5】 前記所定のイベントは、複数の異なるユニットの各々に対する一連のイベントを含み、前記マスタカレンダー手段又はスレーブカレンダー手段におけるエントリの各々は、前記ユニットの1つに対する一連のイベントの次のイベントに対応する請求項1～4のいずれか一つに記載のスケジュールリング回路。

【請求項6】 前記カレンダー制御手段は、前記マスタカレンダー手段におけるエントリをシーケンシャルに処理し、そして処理される各エントリに対し、対応するイベントを生成し、処理したエントリを削除し、そしてマスタカレンダー手段又はスレーブカレンダー手段の1つ

に前記関連するユニットの一連の次のイベントに対応する新たなエントリを作成する請求項5記載のスケジュールリング回路。

【請求項7】 前記カレンダー制御回路によって現在処理中の前記マスタカレンダー手段の記憶ロケーションの1つを識別するアクティビティポインタを有し、そして対応するタイムスロットが前記現時間であるマスタカレンダー手段の記憶ロケーションの1つを識別するタイムポインタを有する請求項6記載のスケジュールリング回路。

【請求項8】 前記アクティビティポインタは現在処理中の記憶ロケーションで、前記エントリの処理、もし1つ以上の場合は前記エントリの全ての処理の終了時に更新され、そして前記タイムポインタは前記現在時間に従って更新される請求項7記載のスケジュールリング回路。

【請求項9】 前記複数のエントリの前記ユニットはそれぞれ自身個々に対応するパラメータブロックを有し、前記パラメータブロックは選択的に前記マスタカレンダー手段におけるエントリ又は前記スレーブカレンダー手段におけるエントリを構成する請求項5～8のいずれか一つに記載のスケジュールリング回路。

【請求項10】 さらに、前記マスタ及びスレーブカレンダー手段におけるエントリとは独立に、前記カレンダー制御手段によってアクセスされるエントリ形式のスヌープメモリを有し、前記スヌープメモリは前記ユニットの1つに対応するパラメータブロックが前記マスタカレンダー手段におけるエントリ又は前記スレーブカレンダー手段におけるエントリを現在構成しているか否かを指示する請求項9記載のスケジュールリング回路。

【請求項11】 前記イベントの複数の異なる優先レベルの各々に対する前記マスタカレンダー手段及びスレーブカレンダー手段を有し、

各優先レベルは他の優先レベルの現在時間とは独立したそれ自身の現在時間を有し、もし前記所望のスケジュールリング時間がその優先レベルに対する現在時間の後方又はそれに等しいイベントに対応したエントリをより高い優先レベルのマスタカレンダー手段が有する場合、前記カレンダー制御手段は特定の優先レベルに対するマスタカレンダー手段でのエントリ処理を停止する請求項1～10のいずれか一つに記載のスケジュールリング回路。

【請求項12】 前記スレーブカレンダー手段は、各第1順位及び第2順位のスレーブカレンダーを有し、前記カレンダー制御手段は、前記イベントスケジュールリング中に、もし前記現在時間とそのイベントの所望のスケジュールリング時間との間隔が前記マスタカレンダースケジュールリング範囲よりも大きな閾値を越えた場合に、第2順位スレーブカレンダーのイベントに対応したエントリを作成し、そして前記間隔が前記マスタカレンダースケジュールリング範囲外であるが前記閾値を越えない場合には、第1順位スレーブカレンダーのイベントに対応

したエントリを作成し、

前記カレンダー制御手段は、また第1順位及び第2順位のスレーブカレンダーをモニタし、その対応イベントが前記閾値よりも近くなった時点で第2順位スレーブカレンダーのエントリを第1順位のスレーブカレンダーへ転送し、そしてその対応イベントがマスタカレンダースケジューリング範囲内となった時点で第1順位スレーブカレンダーのエントリを前記マスタカレンダー手段へ転送する請求項1～11のいずれか一つに記載のスケジューリング回路。

【請求項13】 所定のイベントをスケジュールするスケジューリング方法は、

イベントがスケジュールされる時、現時間とそのイベントの所望のスケジューリング時間との間隔が所定のマスタカレンダースケジューリング範囲内の場合にはそれに対応したエントリがマスタカレンダー手段に作成されること、

もし前記間隔が前記所定のマスタカレンダースケジューリング範囲外の場合には前記イベントに対応するエントリが前記マスタカレンダー手段とは独立にスレーブカレンダー手段に作成されること、

前記スレーブカレンダー手段におけるエントリはモニタされること、

その対応するイベントが前記マスタカレンダースケジューリング範囲内になったエントリは前記マスタカレンダー手段へ転送されること、を特徴とするスケジューリング方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明はスケジューリング回路及び使用方法に関し、例えば、非同期転送モード(ATM)通信ネットワークにおけるセル伝送動作のスケジューリングに関するものである。

【0002】

【従来の技術】ATM通信ネットワークにおいては、ユーザデータや制御情報を異なるネットワークユニット間で伝送するために、複数の“仮想チャネル(VC)”が使われる。そのようなユーザデータ及び制御情報はATMセル形式で転送され、各セルはセル制御及びアドレス情報を含むヘッダ部と、ユーザデータや他の制御情報を含むペイロード部とからなる。

【0003】所定のネットワークユニットは、それを1つ又はより多くのネットワークユニットに接続する複数の仮想チャネルを有する。従って、各ネットワークユニットにはトラフィックマネージャが必要となり、それが関連する異なった仮想チャネルへ与えられるセルの生成時間を管理する。

【0004】トラフィックマネージャのタスクは、異なる形式のトラフィックを伝送するのに異なる仮想チャネルが使用され、それには異なる優先度を有するトラフィッ

ク形式が含まれるため複雑である。例えば、次のトラフィック形式がATMネットワークにおいて使用される。

【0005】i) 固定ビット速度(CBR)トラフィック。これは、典型的には、ビデオ信号のような高優先度データに使用される。

ii) 可変ビット速度(VBR)トラフィック。この形式のトラフィックもまた、比較的高い優先度を有する(しかしながら、CBRトラフィックよりは低い)。幾つかの遅延許容度が利用可能なところ、例えば圧縮ビデオ(モーション・ピクチャ・エキスパート・グループMPEG2)トラフィックや音声トラフィック、で使われる。

【0006】iii) 利用可能ビット速度(ABR)トラフィック。ABRトラフィックを転送するため使用する仮想チャネルは、そのチャネルユーザとネットワークオペレータの間で合意したトラフィックパラメータを有する。合意したパラメータにより課せられる制限の範囲内において、ABRチャネルはネットワークの空き容量を利用するためその伝送速度を可変とすることが許される。ABRトラフィックは低い優先度を有する(CBR及びVBRトラフィックよりも低い)。ABRチャネルは、例えばファイルやデータの伝送に使用される。

iv) 未定義ビット速度(UBR)トラフィック。例えば、Eメールや幾つかのファイル及びデータ伝送等で使用される。

【0007】図1は、ATMネットワークユニットからのセルの伝送時間を管理するために先に提案された“カレンダー(calendar)”スケジューリング方法を説明するためのものである。図1において、カレンダー1は各々連のタイムスロットに対応する記憶ロケーション2のリストから成る。各タイムスロットの所要時間は、ATMネットワークの最大要求データ速度で1つのATMセルを転送するのに要する時間に等しい。

【0008】各記憶ロケーション2は1つ以上のエントリを記憶することが可能であり、各エントリは特定の仮想チャネルがその記憶ロケーションの対応するタイムスロットのトラフィックマネージャによって提供されるべきことを指示する。例えば、カレンダー1の記憶ロケーション2_Aは、仮想チャネルVC_wが記憶ロケーション2_Aと対応するタイムスロットのトラフィックマネージャによって提供されるべきことを示す1つのエントリを有する。

【0009】1つ以上のVCを同じ記憶ロケーションに入力(enter)することが可能である。例えば、カレンダー1の記憶ロケーション2_Bはその中に入力された3つの仮想チャネルVC_x、VC_y及びVC_zを有し、これらの各チャネルが記憶ロケーション2_Bに対応するタイムスロットで提供されるべきことを指示する。記憶ロケーション2_Bの3つのエントリはリンクリスト形式で作成され、タイムスロット当たり大容量のメモリをリザー

5

ブすることを回避し、特定の記憶ロケーションに入力可能なVC最大可能数を収容する。

【0010】トラフィックマネージャは、カレンダー1を処理するのに2つのポインタ3及び4を使用する。第1のポインタ3はカレントタイムポインタ(T)であり、実時間を表す。カレントタイムポインタTは、各セル間隔の後、常にインクリメントされる。第2のポインタ4はアクティビティポインタ(A)であり、Tポインタ3を追い越すことなく、それに常に追いつこうとする。

【0011】カレンダー1の各記憶ロケーションでは、VCが存在しなければAポインタが次の記憶ロケーションへ単にインクリメントされる。この場合、AポインタはTポインタにキャッチアップしようとする。一方、もし一つ以上のVCがAポインタが示す記憶ロケーションに存在する場合、全てのVCが処理されるまでこのポインタはインクリメントされない(この場合、全てのVCを処理するのに必要な時間中、Tポインタは一回以上インクリメントされ、その結果AポインタはTポインタよりかなり遅れることになる)。

【0012】トラフィックマネージャはカレンダー内のエントリに次のサービスを行う。第1に、エントリにおいて指定されたVCのセルが転送される。第2に、もし必要なら、このチャンネルの新たなインターセル(inter-cell)間隔が計算される(CBRチャンネルは常に同じインターセル間隔を有するが、VBR、ABR及びUBRについてはセル間隔はチャンネルの使用によって変化する。そのため、時々計算し直す必要がある)。指定されたVCの次のセル伝送は、カレンダー1の異なる記憶ロケーション2内のVCに新たなエントリを作成することによって再スケジューリングされる。

【0013】例えば、もしVC_xが、関連するATMネットワークユニットからの利用可能な伝送容量の33%を有するならば、ロケーション2_Bでサービスされる時に、カレントタイムポインタTから3セルタイムスロット前方の記憶ロケーション2_Cで再スケジューリングされることになる。

【0014】ところで、本例において、現時間で記憶ロケーション2_Bを示しているAポインタはTポインタの5つ後方の記憶ロケーションであることが分かる。2つのポインタ間の記憶ロケーションに空きがあるため、AポインタはTポインタに対して追いつくことも可能である。しかしながら、A及びTポインタ間でサービスを行うには3つの仮想チャンネルVC_y、VC_z、及びVC_wが存在する。このことは、実際にVC_xが次の処理に再び遭遇する前に3セル期間以上あることを意味する。

【0015】

【発明が解決しようとする課題】図1に示したカレンダースケジューリング方法の1つの問題は、大きく異なる伝送速度を有するチャンネル同士が協働可能なことを要求される点である。最大データ速度155MbpsのA

6

TMネットワークにおける各セル期間は2.726μsである。しかしながら、ある形式のVCは非常に遅いセル速度を有する。例えば、ABRチャンネルは制御のためにリソース管理(RM)セルを使用するが、そのようなRMセルのセル速度は1秒当たり10セル程度である。このことは、図1のカレンダー1が35000エントリ(すなわち、35000×2.726μs=0.1s)を備えることを要求する。これでは、カレンダー1が各タイムスロットのための記憶ロケーション2を必要とすることから、非実用的である。

【0016】この問題に対する1つの解決策は、先に提案されており、各VCにカウンタを持たせることである。このカウンタはある正の値に初期設定され、トラフィックマネージャによる各VCエントリの検査時、カレンダーに関する各パス上でデクリメントされる。VCはそのカウンタがゼロになった時だけサービスされる(すなわち、1つのセルが転送され、次のセル転送がスケジュールされる)。このように、VC当たり6ビットのカウンタを使用して、カレンダーのサイズを(例えば)1000記憶ロケーションに低減可能である。セル速度が370セル/秒(155Mbpsで2.7マイクロ秒のインターセル期間)より大きなVCは、常にカレンダーで“ヒット(hit)”する。すなわち、各VCのカウンタは永久にゼロである。

【0017】上述した解決策は、部分的にはカレンダーサイズを低減するのに有効である。しかしながら、たとえ特定のVCはカレンダーを通した各パス(pass)上で遭遇しても、各遭遇に関してサービスされることを必ずしも意味しない点でトラフィックマネージャのリソースを浪費する。従って、しばしばエントリはカレンダーのこの特定パス上でのサービスを要求しないトラフィックマネージャによって読み出される。

【0018】別の欠点は、フォワードポインタと同様に、カレンダーの各エントリに必要な(リンクリストの先のエントリへの)バックワードポインタを備えることと関連する。なぜなら、各VCはそれ自身のカウンタを持ち、リンクリストは初めから最後へと処理され、各カウンタがゼロのVCはもはや必要とされないからである。カウンタが非ゼロであるリンクリストの残りのVCは、次のカレンダーパスのリンクリストのため保持しておく必要がある。カウンタがゼロになったVCを他のVC(非ゼロカウンタ)を保持しながらリンクリストから削除可能とするには、削除されるVCのバックワードポインタを使用してリンクリストにおける直前のVCを識別し、それによってその直前のVCのフォワードポインタが、削除されるVCのフォワードポインタと等しくなるようにすることが必要である。その結果、削除されるVCに関するリンクリストは再構築される。VC毎のバックワードポインタを提供する必要から、リンクリストの処理が複雑となり、トラフィックマネージャがたくわえ

なければならない情報量が大きく増大する。

【0019】そこで本発明の目的は、上記種々の問題に鑑み、特別な処理パワー及び／又は記憶容量を必要とせず、異種速度のVCを効果的に制御可能な、改良したトラフィックマネージャ及び改良したトラフィック管理方法を提供することにある。

【0020】

【課題を解決するための手段】本発明の第1の態様によれば、所定のイベントをスケジューリングするためのスケジューリング回路は、所定のマスタカレンダースケジューリング範囲内で発生するイベントにそれぞれ対応したエントリを保持するマスタカレンダー手段と、このスケジューリング範囲外で発生するイベントにそれぞれ対応したエントリを保持するスレーブカレンダー手段と、イベントがスケジューリングされる時、現時間とそのイベントの所望のスケジューリング時間との間隔がスケジューリング範囲を超える場合に、スレーブカレンダー手段にそれに対応したエントリを作成するカレンダー制御手段とを有し、カレンダー制御手段は、さらにスレーブカレンダー手段のエントリをモニタし、エントリに対応するイベントがスケジューリング範囲内にあるエントリをマスタカレンダー手段に転送することを特徴とするスケジューリング回路が提供される。

【0021】また本発明の第2の態様によれば、所定のイベントをスケジューリングするスケジューリング方法は、イベントがスケジューリングされる時、現時間とそのイベントの所望のスケジューリング時間との間隔が所定のマスタカレンダースケジューリング範囲内の場合に、それに対応したエントリがマスタカレンダー手段に作成されること、もしその間隔が所定のマスタカレンダースケジューリング範囲外の場合にはイベントに対応するエントリがマスタカレンダー手段とは独立にスレーブカレンダー手段に作成されること、スレーブカレンダー手段におけるエントリはモニタされること、その対応するイベントがマスタカレンダースケジューリング範囲内となったエントリはマスタカレンダー手段へ転送されること、から成るスケジューリング方法が提供される。

【0022】

【発明の実施の形態】図2は、本発明を実施したスケジューリング回路部分を示したものであり、その原理を説明するためのものである。図1を参照して述べたこれまでのスケジューリング回路に対し、図2のスケジューリング回路は、それぞれ一連のタイムスロットに対応した複数の記憶ロケーション2から成るマスタカレンダー1を有する。各記憶ロケーション2は、1つ以上のエントリを持つことが可能で、各エントリはその記憶ロケーションに対応したタイムスロットでサービスされる仮想チャネル(VC)を規定する。図1の回路のように、マスタカレンダー1は、各セル期間でインクリメントされるカレントタイムポインタ(T)3と、現在サービスされ

ている記憶ロケーション2(又は、それとは別に、サービスされる次の記憶ロケーション)を規定するカレントアクティビティポインタ(A)4とを有する。Aポインタは、常時Tポインタに追いつこうとする。特定の記憶ロケーションにエントリが無い時には追いつく機会が生じるが、1つ以上のVCが記憶ロケーション2に入力されるとTポインタの後方に遅れる。

【0023】図2に示すスケジューリング回路では、“スレーブ”と呼ばれるカレンダー6が与えられ、この点で図1のスケジューリング回路と異なる。このスレーブカレンダー6は、複数の記憶ロケーション7と、現在サービス中のスレーブカレンダー6の記憶ロケーション7の1つを示すスレーブポインタ(S)8とを有する。マスタ及びスレーブカレンダー及び関連するA、T及びSポインタはカレンダー制御回路9によって制御される。

【0024】図2のスケジューリング回路は、以下のように動作する。マスタカレンダー1の記憶ロケーション2は、現在サービス中の記憶ロケーションを識別するため、Aポインタ4を使いカレンダー制御回路9によって順次サービスされる。従来と同様に、Aポインタによって指示される記憶ロケーションの各エントリがサービスされると、第1にそのエントリで指定されるVCのセルが転送される。第2に、既存の技術を使って、指定されたVCの次のセルが転送されるべき時間(以降では、“次のスケジューリング時間”又はNSTという)が計算される。そして、図1の回路とは異なる再スケジューリング動作が実行される。この異なる再スケジューリング動作において、カレンダー制御回路は指定のVCのNSTを調べ、マスタカレンダーのスケジューリング範囲SR内か否かを判断する。本明細書において以降で詳細に説明するように、このスケジューリング範囲SRは記憶ロケーションの現在の数未満でマスタカレンダーの記憶ロケーション2の数に比例し、Tポインタがその記憶ロケーション数を縦断するのにかかる時間を表す。

【0025】もし、NSTがマスタカレンダー1のスケジューリング範囲SR内ならば、図1のスケジューリング回路のように、指定されたVCのための新たなエントリがマスタカレンダーの適当な1つの記憶ロケーションに作成される。一方、もしNSTがマスタカレンダー1のスケジューリング範囲SR外ならば、指定されたVCはマスタカレンダー1に入力されない。その代わり、スレーブカレンダー6の記憶ロケーション7の1つに入力され、その記憶ロケーションはまた指定されたVCのNSTを記憶するのに使われる。

【0026】カレンダー制御回路9は、スレーブポインタSを使ってスレーブカレンダー6の記憶ロケーション7を連続的に巡回し、現在処理されている記憶ロケーションを識別する。次に、スレーブカレンダー記憶ロケーション7が特定のVCに対応するアクティブ(又は“ビジ

10

20

30

40

50

一) エントリを含むことを検出すると、そのVCに記憶したNSTを調べる。NSTはマスタカレンダーのカレントタイムポイントTと比較される。もし、NSTと現時間(Tポイント)の間の時間間隔がマスタカレンダーのスケジューリング範囲SRより小さい場合、スレーブカレンダー記憶ロケーション7で指定されるVCはマスタカレンダーへ転送される。すなわち、そのVCに対応する新たなエントリがマスタカレンダー記憶ロケーション2の適当な1つに作成される。

【0027】もし、調べたスレーブカレンダー記憶ロケーション7の記憶したNSTと現時間との間の時間間隔がマスタカレンダーのスケジューリング範囲SRより大きければ、スレーブカレンダーの記憶ロケーションは変更されず、スレーブポイントSのスレーブカレンダー記憶ロケーションによる次のパス完了が保留される。なお、スレーブカレンダーからマスタカレンダーへの不要なVC転送を確実に防止するためには、スレーブポイントSがスレーブカレンダー6の1つのパスを完了するのにかかる時間をマスタカレンダーのスケジューリング範囲SRより小さいか又は等しくする必要がある。

【0028】図2のスケジューリング回路において、マスタカレンダーのサイズは図1のスケジューリング回路のカレンダー1よりもかなり小さくできる。例えば、マスタカレンダー1のサイズは設定済みの全VC数(いかなる所定時間においても、設定チャンネルの全てがアクティブというわけではない)と単純に等しくすることが可能である。例えば、30のそのような設定チャンネルが存在する場合、マスタカレンダーのサイズは図1の回路のカレンダー1と比べて顕著に減少できる。図1の回路のカレンダー1では、さらにVCあたりカウンタを伴い、典型的には1000個の記憶ロケーションを有する。実際、異なるアプリケーションには設定すべき異なる数のVCが必要である。例えば、いくつかのアプリケーションでは1000、4000又は64000個のVCが存在し得る。スレーブカレンダー6については、いかなる時でも存在する設定VCと同数の多くの記憶ロケーション7が必要となる。無論、再スケジューリングにおいてスレーブカレンダーへ転送されることのない非常に高い伝送速度のVCが存在する。この場合には、スレーブカレンダーにおける記憶ロケーション7の数を全てのアクティブなVCの数以上にまで減少させることが可能である。

【0029】また、図2のスケジューリング回路は、先に述べた各VCにそれ自身のカウンタを持たせることに関する欠点を解消するのに有効である。第1に、VCがマスタカレンダーに入力される時はいつも次にサービスされる必要があることから、マスタカレンダーの全てのエントリで“ヒット”が保証される。第2に、マスタカレンダー1の同じ記憶ロケーション内に複数のエントリがリンクリストを使って作成される場合、各エントリに

図1のカウンタベースの手法で用いる種類のバックワードポイントを与える必要がない。なぜなら、その特定のマスタカレンダー記憶ロケーション2がサービスされる時に各リンクリストは一回だけ使用されるからである。サービスの後、そのリンクリストに属する各エントリはマスタカレンダーの異なる記憶ロケーションか又はスレーブカレンダー6の適当な記憶ロケーションのいずれかに再スケジューリングされる(より多くのデータが送信可能のように与えられる)。削除されたエントリに関するリンクリストは再構築の必要がない。このことは、リンクリストの操作を非常に簡潔且つ高速にする。

【0030】次に、本発明の第1の実施例について図3～9を参照して説明する。先ず図3を参照して、第1の実施例のスケジューリング回路は、マスタカレンダー1、各T、A及びSポイント3、4及び8、そしてカレンダー制御回路24からなる。Tポイント3は、2.726 μ s(155Mbpsの伝送速度での1セル期間に相当)の細分化を伴う10.23秒(ABRチャンネルの許可セル速度(ACR)減少時間要素(ADTF)タイムアウト値に相当)を測るのに必要な22ビットのディジタル値である。

【0031】なお、可能な形態として、2.831 μ sの異なるセル期間を使用してもよい。このセル期間は以下のように実現される。先ず、セル/秒で155.52Mbps(各セルは各々8ビットの53バイトを有する)―これは先に示したように2.726 μ sに相当する―のビット速度を表すことによって最大要求データ速度が計算される。そして、このセル速度をSTS3cフレーム(すなわち、全体で270バイトの内の260データバイト―各STSフレームは260データバイトと10オーバーヘッドバイトを含む)のデータに比例して表される要素と乗算する。従って、セル期間は以下の式で表される。

$$(8 \times 53) / (155.52 \times 10^6) \times (270 / 260) = 2.831 \mu s$$

【0032】Aポイント4は16ビットのパラメータである。従って、Aポイントの範囲は、 $2^{16} \times 2.726 \mu s = 0.1787 s$ まで対応する。これは略10セル/秒の最低セル速度(RMセル速度)を扱うのに十分である。Aポイントのサイズは、マスタカレンダーの動的な範囲($1024 (= 2^{10})$ VCに対し10ビットで十分であるが、16ビットによりNST(タイムスタンプ)が 2^{16} セル期間の範囲で可能となる)を決定する。

【0033】本実施例において、記憶ロケーション14の一組12はマスタカレンダー及びスレーブカレンダーでエントリを作成するのに使用される。これらの記憶ロケーション14は、関連するATMネットワークユニットの異なるVCにそれぞれ対応する。この理由から、記憶ロケーション14はこれ以降においてVC毎のパラメータブロック(per-VCパラメータブロック)とし

て述べる。

【0034】図4を参照すると、VC毎のパラメータブロック14は3つの異なる形式に分割される。第1は、マスタカレンダー1にエントリを作成するブロックを有する。図1を参照して先に説明したように、マスタカレンダー1の記憶ロケーション2は、その中に1つ以上のエントリを持つことが可能で、その記憶ロケーションに対応したタイムスロットでサービスされるVCの数に依存する。従って、各マスタカレンダー記憶ロケーション2は各々の開始及び終了ポイントを有する。開始ポイント10は関連する記憶ロケーション2の最初のエントリを構成するVC毎のパラメータブロック14を指示し、そして終了ポイントは関連する記憶ロケーション2に入力された最後のVC毎のパラメータブロックを指示する。もし、記憶ロケーション2に入力されたVCが1つだけの場合には、開始及び終了ポイントは同じ値であり、両者とも1つのVCに対応するVC毎のパラメータブロックを指し示す。

【0035】1つより多くのVCが特定のマスタカレンダー記憶ロケーション2に入力された時には、VC毎のパラメータブロックは全てのエントリのリンクリストを構成するのに使われる。この目的のために、各VC毎のパラメータブロックは“NEXT_VC”フィールドを有し、そこにはマスタカレンダー記憶ロケーション2に入力される次のVCのVC毎のパラメータブロックのメモリロケーションを示すポイント値が含まれる。リンクリストにおける最後のVC毎のパラメータブロックのNEXT_VCフィールドは“don't care”状態である。

【0036】例えば、図4の例で示すように、記憶ロケーション2Aは内部に入力された4つのVCsである。VC2、VC5、VC8、及びVC3をこの順序で有する。従って、開始ポイントはVC2を指し示す。VC2のNEXT_VCフィールドはVC5を指し示す。順次、VC5のNEXT_VCフィールドはVC8を指示する。VC8のNEXT_VCフィールドはVC3を指示する。VC3のNEXT_VCフィールドはそれが記憶ロケーション2Aに入力された最後のVCであることから、“don't care”に設定される。マスタカレンダーの終了ポイントはまたVC3を指示する。

【0037】VC毎のパラメータブロック14の第2の形式は、スレーブカレンダーエントリである。このブロックは図4において斜線で示してある。この本例では、VC1、VC6、及びVC7がスレーブカレンダーエントリとして現在使用中である。これらのパラメータブロックは、関連するVCによって1つのセルを転送するための次のスケジュール時間NSTを記憶するためNSTフィールドを用いる。NSTフィールドは、16ビットパラメータとしてNSTを記憶する。VC毎のパラメータブロックの第3の形式は未使用ブロックであり、図4

ではVC4を例として示している。

【0038】図4に示すように、VC毎のパラメータブロック14の第1及び第2の両形式ともそれぞれ“QUEUE_START”フィールドを含んでいる。スケジューリング回路を含むATMネットワークユニットの動作中、スケジューリング回路によって転送されるセルデータはネットワークユニットの別の回路によって連続して生成される。セルデータは待ち行列（キュー）を作ってVC毎の原則で記憶される。従って、VC毎のパラメータブロック14のQUEUE_STARTフィールドはそのセルデータキューの先頭を指し示す。セルが送出された後、QUEUE_STARTフィールドは新たなセルデータキューの先頭を指示するように更新される。なお、スレーブカレンダーエントリを与えるのに使われるVC毎のパラメータブロックは、スレーブカレンダーのVCがリンクリスト中で互いにリンクされないように、それらの各NEXT_VCフィールドを“don't care”状態にする。

【0039】図3に戻って、第1の実施例のスケジューリング回路はさらにそれぞれスレーブ及びマスタのスヌープ(snoop)メモリ16及び20を有する。まず、スレーブスヌープメモリ16の処理を取り扱うために、一組12のVC毎のパラメータブロック14はNブロック（本例ではN=32）のグループに分割される。スレーブスヌープメモリ16はNビット幅を有する。従って、スレーブスヌープメモリ16の各Nビットワード18は組12のN連続したVC毎のパラメータブロック14の1つのグループと対応し、Nビットワード18の各ビットはそのワード18に対応するグループのパラメータブロック14の1つと個々に対応する。1つのビットが1に設定された時、対応するVC毎のパラメータブロック14がスレーブカレンダーエントリとして使用されることを意味する。そのビットがリセットされると、それとは反対に、対応するパラメータブロック14がマスタカレンダーに入力されるか又は全く未使用であることを意味する。

【0040】スレーブスヌープメモリ16により、カレンダー制御回路24は各パラメータブロックを直接読み出す必要もなしにスレーブカレンダーエントリを構成するこれらのパラメータブロック14を識別することが可能になる。スレーブポイントSはマスタカレンダー1のスケジューリング範囲SRより決して大きくならない時間でVC毎のパラメータブロック14の組12のパス(pass)を完了する必要がある。このことは、記憶に留めるべき重要な点である。

【0041】マスタスヌープメモリ20は、マスタカレンダー1に関して同様な目的で使用される。ここでも、マスタカレンダーはN個の連続した記憶ロケーション2（本例でもN=32）のグループに分割される。マスタスヌープメモリ20はNビット幅を有し、各Nビットワ

ワード22はN個の連続した記憶ロケーション2の1つのグループに対応する。Nビットワード22内の各ビットは、個々にそのグループの1つの記憶ロケーション2に対応する。この場合、ワード22の1ビットが1にセットされた時、対応する記憶ロケーション2はその中に入力された少なくとも1つのVCを有することを意味する。もし、このビットがゼロならば、反対に、対応する記憶ロケーション2は“空”、すなわち有効なエントリを含まない、ことになる。

【0042】マスタスヌープメモリ20は、カレンダー制御回路24によっていかなるVCエントリをも含まない記憶ロケーション2の検査を回避するために使用される。これは、Aポインタ4がTポインタ3に追いつくことを可能とする点で有用である。マスタカレンダーヌープ20の具体的な使用例を図5及び6を参照して説明する。

【0043】図5には、マスタカレンダーヌープメモリ20の特定のワード22_Aに対応した32個のマスタカレンダー記憶ロケーション2のグループが示されている。このグループの記憶ロケーションは、物理的な記憶ロケーション0020（16進）～003F（16進）を有する。ワード22_Aにおいて、最初の非ゼロビットは最下位のビット（LSB）の左側6番目に位置し、関連グループの最初のカレンダーエントリは0020+6=0026（16進）の位置であることを意味する。

【0044】その結果、カレンダー制御回路24は、ワード22_Aを読むことによって0020～0025（16進）の記憶ロケーションをサービスする必要がないことを知る。図6に示すように、処理すべき最初の記憶ロケーションは0026（16進）である。なお、この記憶ロケーション（本例では0026）はカレントタイムポインタTの前方であってもよい。この場合、カレンダー制御回路24は、Tポインタが記憶ロケーション0026のエントリにサービスを行う前、そのロケーションに達するまで待つ。そのような待ちを要求できるのは、さもなければ、Tポインタの前方の記憶ロケーションが許可されていないのにサービスを受ける可能性がおきるからである。

【0045】先に述べたように、アクティビティポインタAは、どんな特定のタイムスロットでもスケジューされたVCを超えてよい。カレントタイムポインタTの後方に遅れる可能性がある。マスタカレンダー1は連続したループ中で処理がなされるため（すなわち、A及びTポインタは、マスタカレンダーの最後の記憶ロケーションに達した後、その最初の記憶ロケーション2へ“ラップアラウンド（wrap around）”するか又は戻る）、マスタカレンダーの範囲近傍にスケジューされているチャンネルが、アクティビティポインタAのまだサービスされていない、従ってそのチャンネルデータフローを破壊し得るマスタカレンダーの一部へ書き込まれない

ようにする保護機構が必要になる。

【0046】そのような保護機構を与えるため、マスタカレンダーのスケジューリング範囲SRは、記憶ロケーションのプリセット数によりマスタカレンダーの全記憶ロケーションの数Lより少なく設定される。もし、マスタカレンダーにおける記憶ロケーションの数が約1000ならば、一例として256記憶ロケーションが設定される。

【0047】図7を参照して、アクティビティポインタAは、マスタカレンダーの記憶ロケーションのある数（256以下）だけタイムポインタTの後方に遅れて示されている。先述したように、Aポインタによって指し示される記憶ロケーションの各エントリがサービスされる時、次のスケジューリング時間（NST）がマスタカレンダー1のスケジューリング範囲（図7の斜線範囲）より前方（すなわち、Tポインタによって測量されるような現時間の前方）のさらに先にある場合、関連するVCをスレーブカレンダーで再スケジューすべきかを判断される。言い換えれば、NSTとカレントタイムポインタTとの間の差がカレンダーのスケジューリングより大きいか否かがチェックされる。この差は、Tポインタがマスタカレンダーを“ラップアラウンド”していても容易に計算できる。なぜなら、NSTはTに関して常に先方にあると仮定し得るからである。

【0048】それとは異なる再スケジューリングの可能性を以下で説明する。もし、計算したNSTが現時刻よりも前ならば（CBRチャネルで可能なように）、図7のTポインタの前方の記憶ロケーションBはNSTに対応し、カレントタイムポインタTによって指示される記憶ロケーションの後の次の記憶ロケーション（T+1）でスケジューされるよう、カレンダー制御回路24はNSTを切り上げる。

【0049】なお、CBRトラフィックはAポインタとの関連でスケジューされるが、もしNSTがTポインタ（それ以下）の前方にあるなら、Tポインタの後（T+1）に置かれなければならない。他の全てのトラフィックはTポインタとの関連でスケジューされる。マスタスヌープメモリはAポインタの前方ロケーションの読み出しを許容できるため、AポインタはTポインタの前を走行可能である。しかしながら、この場合、そのようなロケーションへのいかなるサービスもTポインタがそれらに追いつくまでは実行されない。

【0050】NSTがスケジューリング範囲SR内ならば、図7の記憶ロケーションCはNSTに対応し、カレンダー制御回路24はスケジューリング範囲内で記憶ロケーションCにVCを入力する。もし、記憶ロケーションが先に空であった場合には、そのロケーションの開始及び終了ポインタは再スケジューされるVCに対応したVC毎のパラメータブロックを指し示すように更新される。しかしながら、もし関連する記憶ロケーションに

すでに1つ以上のエントリが存在していた場合、カレンダー制御回路24は記憶ロケーションCにおける既存リンクリストの最後に再スケジュールされるVCに対応したVC毎のパラメータブロック14を付加する。このことは、そのリストに付加されるVC毎のパラメータブロック14のメモリロケーションを指し示すように既存リンクリストにおける最終VC毎のパラメータブロック14のNEXT_VCフィールドの更新と、そして次にこの付加されたVC毎のパラメータブロック14を指し示すようマスタカレンダー記憶ロケーション2の終了ポインタの更新とを伴う。

【0051】もし、NSTがスケジューリング範囲（例えば、図7の記憶ロケーションD）を超えてVCを再スケジュールすべきことを指示していた場合、カレンダー制御回路はマスタカレンダーにエントリを作成することを許可せずに、代わりにスレーブカレンダーにVCを入力する。これは、再スケジュールされるVCに対応したスレーブスヌープメモリ16のビットを1に設定し、そして計算したNSTを反映すべくこのVCに対応したVC毎のパラメータブロック14のNSTフィールドを更新することによって単純に達成される。

【0052】次に図8を参照して、マスタカレンダーエントリをサービスするのにカレンダー制御回路24が実行するステップを説明する。初めに、ステップS1において、カレンダー制御回路24はAポインタで指し示された記憶ロケーションで処理すべき次のエントリを決定する。このエントリは（関連する記憶ロケーションで処理すべき2番目の又は引き続くエントリであっても）、記憶ロケーションの開始ポインタを使って識別される。ポインタは、関連するVCのVC毎のパラメータブロック14のメモリロケーションを指し示す。一旦、処理すべきVCのVC毎のパラメータブロック14が識別されると、そのパラメータブロックのQUEUE_STARフィールドが調べられ、送信すべきセルの位置を決定する。このセルは次に送信され、QUEUE_STARフィールドはこのVCの次のサービスの準備のため更新される。

【0053】記憶ロケーションの開始ポインタもまた、その記憶ロケーションにおける次のエントリ（もしあるなら）を指し示すため更新される。すなわち、開始ポインタは現在処理中のエントリのNEXT_VCフィールドに保持された位置に設定される。そのようなエントリが存在しなければ、すなわちその時点で開始ポインタが終了ポインタと同じ値を持つなら、マスタスヌープメモリ20がマスタカレンダーからその記憶ロケーションをとり除くべく更新される。

【0054】ステップS2では、処理されるVCのNSTを決定するのに従来の処理が適用される。なお、NSTを計算する時、その計算は好適には1つのフルレート（full-rate）セル期間よりも高い精度で、すなわちセル

期間の端数（fraction）を用いて、実行される。この端数は（例えば、NSTフィールドに付加された端数フィールドを使って関連するVCのVC毎のパラメータブロックに）保持され、NSTが計算される各時間毎に使用される。これにより、マスタカレンダーにおけるスケジューリングスロットの各々が1つのフルレートセル期間に対応していたとしても、VCはインターセル間隔がフルレートセル期間の整数に等しくない平均転送速度を実現することができる。ステップS3では、計算したNSTがマスタカレンダーのスケジューリング範囲内か否かが決定される。

【0055】もし、NSTがスケジューリング範囲内ならば、処理はステップS4へ進み、そこでNSTがTポインタと比較される。NSTがTポインタの後方ならば（ステップS4で“yes”）、処理はステップS5へ進み、そこでカレンダー制御回路はNSTを切り上げてTポインタに1を加算した値（T+1）に等しくする。反対に、ステップS4で結果が“no”なら、処理はステップS6へ直接進む。

【0056】ステップS6では、VCはマスタカレンダーで再スケジュールされる。VCが再スケジュールされるマスタカレンダー記憶ロケーションは、NSTに基づいて決定される。その記憶ロケーションが初め空きならば、その開始及び終了ポインタが更新され、処理されるVCに対するVC毎のパラメータブロック14を指し示す。さもなければ、先述したように、VC毎のパラメータブロックが関連する記憶ロケーションの位置で既存のリンクリストに付加される。

【0057】もし、ステップS3の結果が“no”なら、すなわち計算したNSTがマスタカレンダーのスケジューリング範囲SR外なら、処理はステップS7へ進む。ステップS7では、処理されるVCがスレーブカレンダーで再スケジュールされる。初めに、計算されたNSTがそのVCに対応するVC毎のパラメータブロック14のNSTフィールドに記憶される。次に、そのVCに対応したスレーブスヌープメモリ16のビットが1にセットされる。そして、処理はステップS8へ進む。

【0058】最後に、ステップS8で、ちょうど処理したVCがその記憶ロケーション（その記憶ロケーションをマスタカレンダーから削除するためステップS1で更新されたマスタスヌープメモリ20）で処理すべき最後のVCであったか否かが決定される。もし、そうなら、次のビジーロケーションを判断して、次のサービス動作の準備のためにマスタスヌープメモリ20を使ってAポインタが更新され、そして処理は終了する。

【0059】ここでは、スレーブカレンダーの処理を図9を参照して説明する。スレーブカレンダーの処理は、マスタカレンダーの処理と並行して実行される。最初のステップS10では、カレンダー制御回路24は、スレーブスヌープメモリ16を使って、スレーブカレンダー

10

20

30

40

50

における S ポインタ 8 の前方の最初のビジローケーションを決定する。そして、ステップ S 1 1 で、そのビジローケーションで VC 毎のパラメータブロックの NST フィールドを読み出す。ステップ S 1 2 では、NST フィールド中の記憶された NST をカレントタイムポイント T と比較する。記憶した NST と T との間の差がマスタカレンダーのスケジューリング範囲 SR 内である場合は、処理が S 1 3 へ進む。

【0060】なお、図 9 に示すように、記憶した NST は、実際には、T ポインタに所定のオフセット時間 ΔT を加算したものと比較される。記憶した NST と $(T + \Delta T)$ との間の差がマスタカレンダーのスケジューリング範囲 SR 内である場合は、処理は S 1 3 へ進む。オフセット時間 ΔT はマスタカレンダーへの VC 転送に伴うメモリ書き込み動作の時間を許容するのに必要であるため比較で用いる。オフセット時間はゼロに設定してもよい。

【0061】ステップ S 1 3 では、カレンダー制御回路 2 4 が、NST からマスタカレンダーに入力すべき VC の位置（記憶ロケーション 2）を決定する。そして、ステップ S 1 4 で、VC がその記憶ロケーション入力される。このステップ S 1 4 は、基本的に図 8 と関連して先述したステップ S 6 と同様である。次に、ステップ S 1 5 で、この VC に対応するスレーブスヌープメモリ 1 6 のビットがリセットされ、VC に対応するマスタスヌープメモリ 2 0 のビットがセットされる。それによりスレーブカレンダーからマスタカレンダーへの VC 転送が反映され（ステップ S 1 6）、処理が終了する。ステップ S 1 2 では、記憶した NST と T（又は $T + \Delta T$ ）との間の差がマスタカレンダーのスケジューリング範囲 SR を超えることが判明した場合に、アクションなしに処理が終了する。

【0062】図 10 は、本発明の第 2 の実施例によるスケジューリング回路部分を示している。本実施例は異なる優先度のセルを効果的に制御可能とすることを意図している。図 10 の回路は、4 つのマスタカレンダー 1-1 ~ 1-4 と、有効な 4 つのスレーブカレンダー 6-1 ~ 6-4 とを有する。各マスタカレンダー 1-1 ~ 1-4 は第 1 の実施例のマスタカレンダー 1 と同じ構成を有する。第 2 の実施例のスレーブカレンダー 6-1 ~ 6-4 もまた第 1 の実施例のスレーブカレンダー 6 と基本的な構成を同じくする。

【0063】各マスタカレンダーは、それ自身の T ポインタ 3 及び A ポインタ 4 を有する。例えば、マスタカレンダー 1-1 は T ポインタ 3-1 (T1) 及び A ポインタ 4-1 (A1) を有する。各スレーブカレンダー 6 は、それ自身の S ポインタ 8 を有する。例えば、スレーブカレンダー 6-1 はそれ自身の S ポインタ 8-1 (S1) を有する。カレンダー制御回路 3 4 は、全てのマスタ及びスレーブカレンダー 1 及び 6 に接続され、そして

スケジューリング回路の制御動作のために全てのポイント 3、4 及び 8 に接続される。

【0064】図 10 の回路において、各マスタカレンダー 1-1 ~ 1-4 は ATM ネットワークユニット内で異なる優先レベルのセルトラフィックに対応する。例えば、マスタカレンダー 1-1 は最高の優先度を有し、そしてマスタカレンダー 1-2 は次の高い優先度を有する、等である。本明細書の最初の部分で指摘したように、ATM ネットワークの異なる仮想チャネルは、異なる優先度を有する異なる形式のセルトラフィックを伝送する。高優先度の仮想チャネルは一定ビット速度 (CBR) チャネルを含む。図 10 の回路において、従って、そのようなチャネルは最高優先度のマスタカレンダー 1-1 でスケジューリングされる。可変ビット速度 (VBR) チャネルもまた高い優先度を有し（しかしながら、CBR チャネルよりは低い）、マスタカレンダー 1-2 でスケジューリングされる。利用可能ビット速度 (ABR) チャネルは低い優先度を有し、マスタカレンダー 1-3 でスケジューリングされる。最後に、不特定ビット (UBR) 速度チャネルは最低優先度のマスタカレンダー 1-4 でスケジューリング可能である。

【0065】各マスタカレンダー 1 は、それ自身個々に対応したスレーブカレンダー 6 を有する。例えば、スレーブカレンダー 6-1 は、図 10 のマスタカレンダー 1-1 に対応する。各マスタカレンダー 1 及びそれに個々に対応したスレーブチャネルは、基本的には上述した第 1 の実施例と同様に動作する。特に、マスタカレンダーはそれぞれの VC に対応したエントリを記憶するように与えられる記憶ロケーション（図 10 には示していない）を有する。これらの記憶ロケーションは、関連するマスタカレンダーの A ポインタが進む時に、カレンダー制御回路 3 4 によって連続してサービスされる。関連するマスタカレンダーの T ポインタは各セル期間ごとに進み、A ポインタは典型的には T ポインタの項方に遅れることになる（しかしながら、常時追いつこうとする）。

【0066】A ポインタによって指し示された記憶ロケーションのエントリがサービスされる時、そのエントリで特定される VC セルが転送され、その VC は再スケジューリングされる。そのような再スケジューリングでは、もし VC の次のスケジューリング時間 NST がマスタカレンダーのスケジューリング範囲 SR 内の場合には、VC はマスタカレンダーの適当に引き続くエントリにおいて再スケジューリングされる。一方、NST がマスタカレンダーのスケジューリング範囲 SR を超える場合には、VC は関連するマスタカレンダーに対応するスレーブカレンダー 6 に入力される。第 1 の実施例と関連して先に述べたように、NST もまたスレーブカレンダーのエントリに含まれる。

【0067】各スレーブカレンダー 6 は、関連する S ポインタによって連続的にスキャンされる。S ポインタに

より関連するスレーブカレンダー6を通したパスの完了にかかる全体の時間は、それが対応するマスタカレンダーのスケジューリング範囲SRより少なく与えられる。Sポインタにより指し示される記憶ロケーションでVCがスレーブカレンダーにおいて発見されると、先に述べたようにVCの記憶したNSTは関連するスレーブカレンダーが対応するマスタカレンダーのカレントタイムポインタTと比較される。もし、VCが現在マスタカレンダーのスケジューリング範囲内にあるならば、そのVCはスレーブカレンダーからマスタカレンダーへ転送される。反対に、VCがいまだにスケジューリング範囲外ならば、そのVCはSポインタによってスレーブカレンダーを通った次のパスのために残される。

【0068】図10には示されていないが簡易化のため、第2の実施例におけるスケジューリング回路が、オプションとして各マスタカレンダー1のマスタスヌープメモリ及び/又は各スレーブカレンダー6のスレーブスヌープメモリに与えられる。実際、セル優先度が認識されるATMネットワークユニットにおける異なるチャネルの競合する要求のため、第2の実施例におけるそのようなマスタ及びスレーブスヌープメモリの使用は好ましいものである。

【0069】次に、第2の実施例（そのようなマスタ及びスレーブスヌープメモリを含む）の動作を図11を参照して説明する。図11は、マスタカレンダー1-1～1-4と関連する動作を示している。各スレーブカレンダー6-1～6-4は、図9で参照したこれより前に述べたフローチャートと自動的に一致して動作する。図11に示すステップシーケンスは各セル期間で実行される。セル期間の開始で、ステップS20では、処理されるマスタカレンダーの優先レベルを表すパラメータPRLは1（最高優先度）にセットされる。そして、ステップS21では、優先レベルPRLを持つマスタカレンダー1のT及びAポインタの間のギャップがある所定の閾値THR（例えば、256）より大きいか否かがチェックされる。大きくなければ、関連するマスタカレンダーのTポインタはステップS22で1インクリメントされる。さもなければ、Tポインタはインクリメントされず、関連するマスタカレンダーは実質的には停止され、AポインタにTポインタに追いつく機会を与える。

【0070】ステップS23で、カレンダー制御回路34は優先レベルPRL（初期1）に対するマスタスヌープメモリを読み出す。ステップS24では、ステップS23で読み出されたマスタスヌープメモリの結果が分析され、Aポインタに即近の範囲に如何なる“ビジー”マスタカレンダー記憶ロケーション、すなわち少なくとも1つのエントリを含む記憶ロケーション、があるか否かが判断される。もしあるなら、カレンダー制御回路34は、ステップS25でこのビジーロケーションがTポインタの前方であるか否かを判断する。ビジーロケーシ

ンがTポインタの前方にない場合には、カレンダー制御回路34は処理すべき次のエントリが現在の優先レベルマスタカレンダーのビジーロケーションにおけるエントリ（又は2以上のエントリなら次のエントリ）かを判断する。

【0071】このエントリは、ステップS26で処理される。ステップS26で実行される処理ステップは、実行的には図8のステップS1～S9に対応する。ステップS24でビジーロケーションがなければ、又はステップS25でTポインタの前方に次のビジーロケーションが発見された場合には、カレンダー制御回路34は、この場合、現在の優先レベルPRL（初期1）でマスタカレンダーをサービスする必要なしと判断し、そしてステップS27で次の最低優先レベルへ移動するために優先レベルパラメータPRLをインクリメントする。ステップS28で、可能な全ての優先レベルがサービスされたか否かを判断する。そうでなければ、処理はステップS21へ戻り、次の優先レベルに対するマスタカレンダーをサービスする。4つの全ての優先レベルが処理されたならば、処理はステップS28の後に終了する。

【0072】このように、図11で示したように、もしある優先レベルに対しマスタスヌープメモリにより指示されるビジーロケーションがなければ、カレンダー制御回路は次の最低優先レベルへ移動し、そのレベルのマスタスヌープメモリを読み出す。同様に、ある優先レベルでマスタカレンダーの次のビジーロケーションがそのレベルのカレントタイムポインタTの前方にある場合には、カレンダー制御回路34は次の最低優先レベルへ移動し、より低い優先度のマスタカレンダーから送信するのに適任のロケーションを探索する。

【0073】第2の実施例において、4つのスレーブカレンダー6-1～6-4が1つのスレーブメモリによって与えられることが可能である。すなわち、メモリの共通エリア内には4つの論理スレーブカレンダーが存在する。この場合、1つのスレーブメモリの各エントリは適当な方法によって4つの論理スレーブカレンダー6-1～6-4の1つに割り当てられる。例えば、各エントリに4つの優先レベルの1つを識別可能な2ビットの優先レベルフィールドが与えられる。この場合、各スレーブカレンダーは（上述したような）そのスレーブポインタを持つことが可能であり、又はシングルスレーブポインタが優先順位により4つの論理スレーブカレンダーの全てをサービスするのに使用可能である。ステップS21をクロックサイクル毎に全ての優先レベルで並列に実行させることで、図11のフローチャートに示す処理がちょうど1クロックサイクルを表すことも可能である。

【0074】図12は本発明によるスケジューリング回路の第3の実施例部分を示す。本実施例では、（第1の実施例におけるような）マスタカレンダー1が存在するが、第1の実施例における1つのスレーブカレンダー6

に代えて、それぞれ第1順位及び第2順位のスレーブカレンダー36及び46が存在する。第1順位のスレーブカレンダー36はSポイント38(1S)を有し、そして第2順位のスレーブカレンダー46はSポイント48(2S)を有する。マスタカレンダー1は通常のT及びAポイント(図12には示されていない)を有する。

【0075】第3の実施例は、次の見地から与えられる。第1の実施例では、スレーブカレンダーの各エントリがマスタカレンダーのスケジューリング範囲より小さいか又は等しい間隔でサービスされる必要がある。しかしながら、スレーブカレンダーで読み出すべきロケーションが多ければそれだけカレンダー制御回路が全てのスレーブカレンダーエントリをスケジューリング範囲より少ないか又は等しい時間で処理するのが困難となる。この困難を緩和するために、第3の実施例ではスレーブカレンダーの“階層化(hierarchy)”を行う。この場合、階層における2つのスレーブカレンダー36及び46は時間の大きさの異なる順位を表す。例えば、マスタカレンダーのスケジューリング範囲SRが1ミリ秒なら、第1順位のスレーブカレンダー36は次の10ミリ秒で送出が必要なセルを有するVCを処理するのに使われ、そして第2順位のスレーブカレンダーは次の100ミリ秒で送出が必要なセルを有するVCを処理するのに使われる。従って、マスタカレンダー1に入力されたVCが処理され次に再スケジュールされる時、もし計算したNSTが10ミリ秒より大きければ、そのVCは第2順位のスレーブカレンダー46で再スケジュールされる。しかしながら、その計算したNSTがマスタカレンダーのスケジューリング範囲(1ミリ秒)より大きく且つ10ミリ秒より小さい場合、それは第1順位のスレーブカレンダー36で再スケジュールされる。先の実施例と同様に、そのNSTがスケジューリング範囲内の場合は、VCはマスタカレンダーで直接再スケジュールされる。

【0076】スレーブカレンダーエントリの処理に関する限り、第2順位のスレーブカレンダー46の場合には、その中の各エントリは10ミリ秒毎に少なくとも一度サービスされる必要がある、その結果VCはそのNSTが10ミリ秒離れていない時に第1順位のスレーブカレンダーへ転送される。第1順位のスレーブカレンダー36の場合には、各エントリが1ミリ秒毎に少なくとも一度サービスされる必要がある、その結果VCはそのNSTがマスタカレンダーのスケジューリング範囲内の時に第1順位のスレーブカレンダーからマスタカレンダーへ転送される。

【0077】従って、セル転送の間隔が10ミリ秒より大きいかなるVCも、第2順位のスレーブカレンダーでその時間部分を費やし、それによって第1順位のスレーブカレンダー上の負荷を軽減する。100ミリ秒のセル転送間隔を有するVCは、第2順位のスレーブカレンダー46に入力された時間の略90%を、第1順位のス

レーブカレンダー46に入力された時間の略9%を費やし、そしてマスタカレンダーに入力された時間に費やす時間は単に1%である。スレーブのカレンダーの順位数は幾つでもよく、カレンダーのサイズ(利用可能なメモリ量)及びトラフィック速度の相違等に基づく。

【0078】本発明の第3の実施例は、特に、大きく異なる範囲のトラフィック速度、すなわち非常に低速なVCや他の非常に高速なVC等、が存在するATMネットワークユニットで使用するのに適している。そのような状況において、第3の実施例はスレーブカレンダーを通したサーチ時間の短縮に顕著な有効性を示す。

【0079】第1順位及び第2順位のスレーブカレンダーのエントリは、マスタカレンダーと同様に、先述したような1組のVC毎のパラメータブロックを使って作成される。この場合、各スレーブカレンダーは、それ自体個々に対応するスレーブスヌープメモリが与えられ、各パラメータブロックが対応するスレーブカレンダーのエントリとして現在使用されているか否かを示す。これにより、スレーブカレンダー間のVC転送を各スレーブスヌープメモリの1ビットを変えることで簡易且つ素早く実行することが可能となる。

【0080】

【発明の効果】以上、本発明をATMセル転送のスケジューリングに関して説明してきた。本発明は、他の実施例において、例えばあるイベント(事象)をスケジューリングする必要がある状況下で提供可能なことは容易に理解されるところである。このように本発明によれば、特別な処理パワー及び/又は記憶容量を必要とせず、異種速度のVCを効果的に制御可能な、改良したトラフィックマネージャ及び改良したトラフィック管理方法が提供可能となる。

【図面の簡単な説明】

【図1】ATMネットワークで用いられる従来のカレンダースケジューリング方法の図式的な説明図である。

【図2】本発明によるスケジューリング回路の基本構成を図式的に示した図である。

【図3】本発明による第1の実施例のスケジューリング回路部分を示した図である。

【図4】第1の実施例である図3の詳細図である。

【図5】第1の実施例におけるマスタスヌープメモリの動作の説明図である。

【図6】マスタスヌープメモリの使用を説明した図である。

【図7】第1の実施例の再スケジューリング動作の説明図である。

【図8】第1の実施例のマスタカレンダーにおけるエントリ処理のフロー図である。

【図9】第1の実施例のスレーブカレンダーにおけるエントリ処理のフロー図である。

【図10】本発明による第2の実施例であって、異なる

23

形式のセルトラフィック処理を可能としたスケジューリング回路部分を示した図である。

【図 1 1】 第 2 の実施例の動作フロー図である。

【図 1 2】 本発明による第 3 の実施例のスケジューリング回路部分を示した図である。

【符号の説明】

1、1-1~1-4…マスタカレンダー

2、7、14…記憶ロケーション

3、3-1~3-4…カレントタイムポインタ

4、4-1~4-4…カレントアクティビティポインタ

6、6-1~6-4…スレーブカレンダー

8、8-1~8-4、38、48…スレーブポインタ

9、24、34…カレンダー制御回路

12…一組の記憶ロケーション

14…VC 毎のパラメータブロック

16…スレーブスヌープメモリ

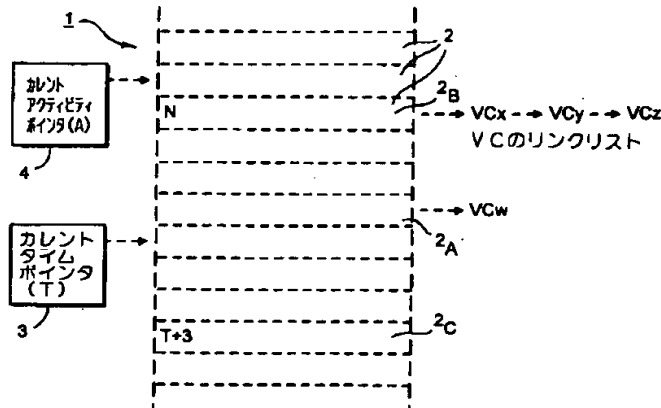
18、22…Nビットワード

20…マスタスヌープメモリ

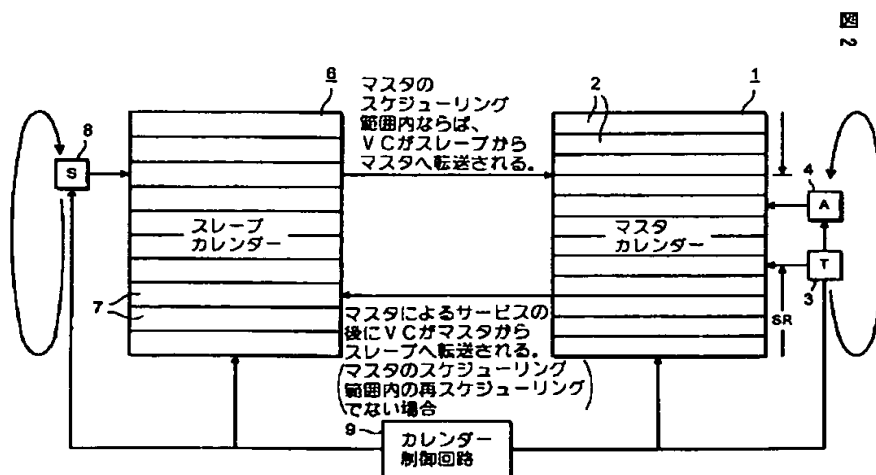
36…第 1 順位のスレーブカレンダー

10 46…第 2 順位のスレーブカレンダー

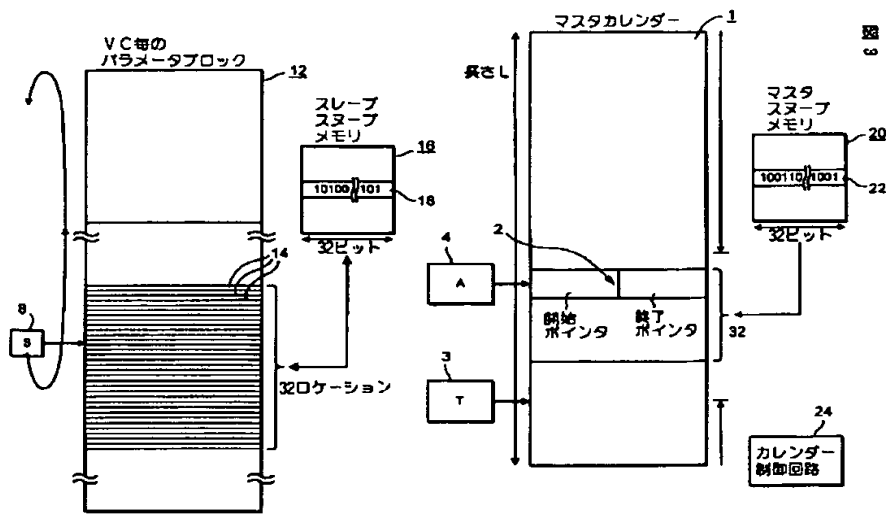
【図 1】



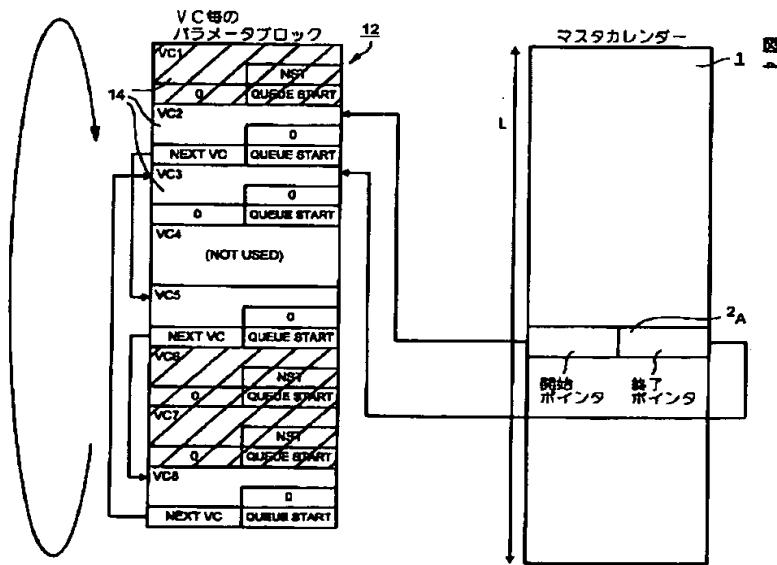
【図 2】



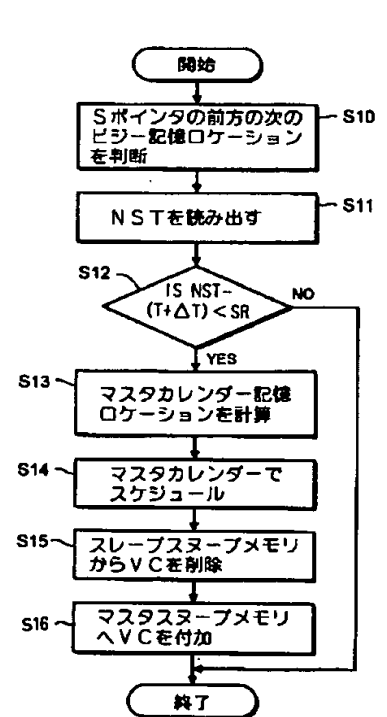
【図 3】



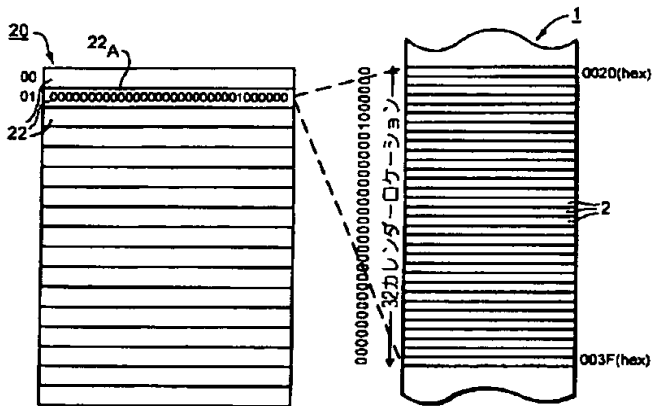
【図 4】



【図 9】

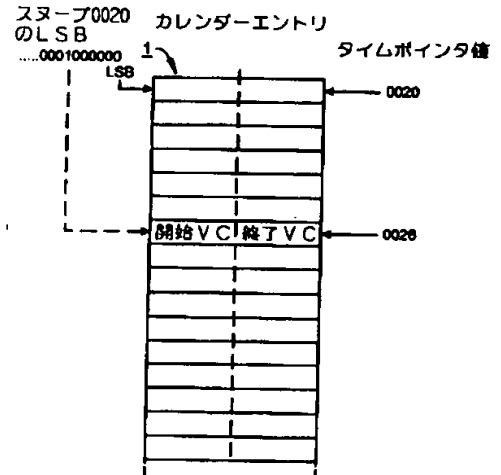


【図 5】



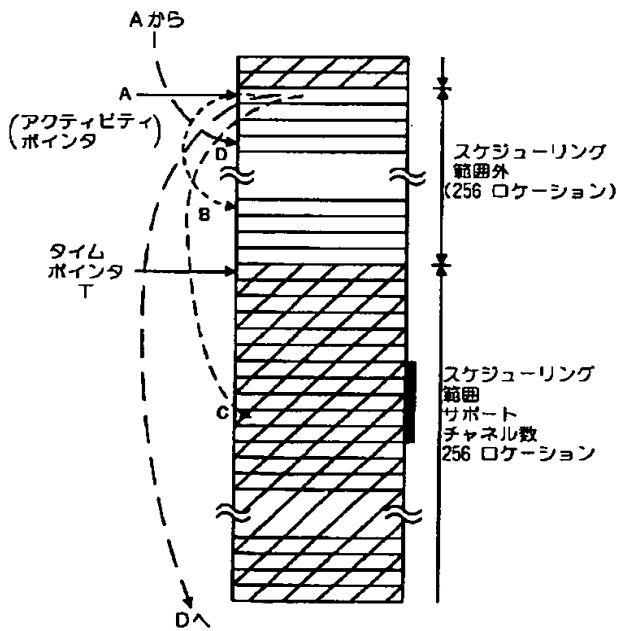
【図 6】

図 6



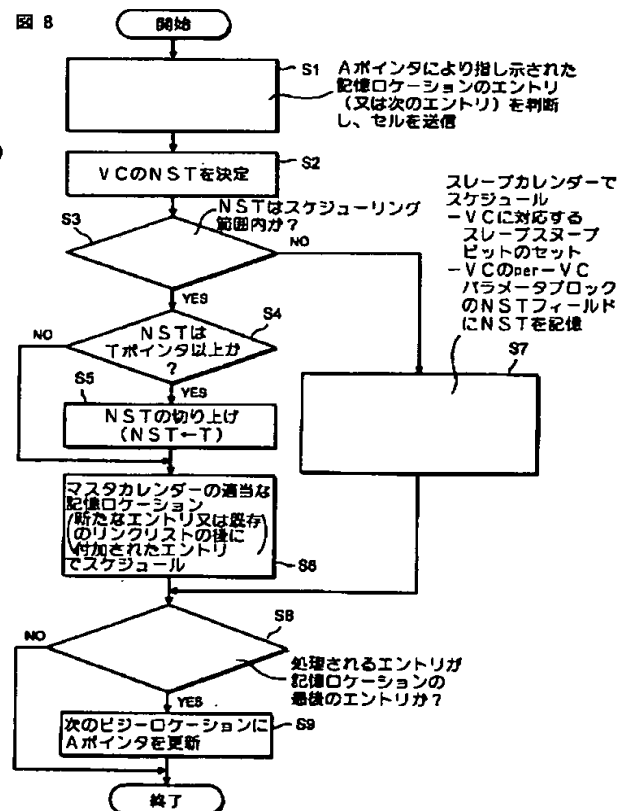
【図 7】

図 7

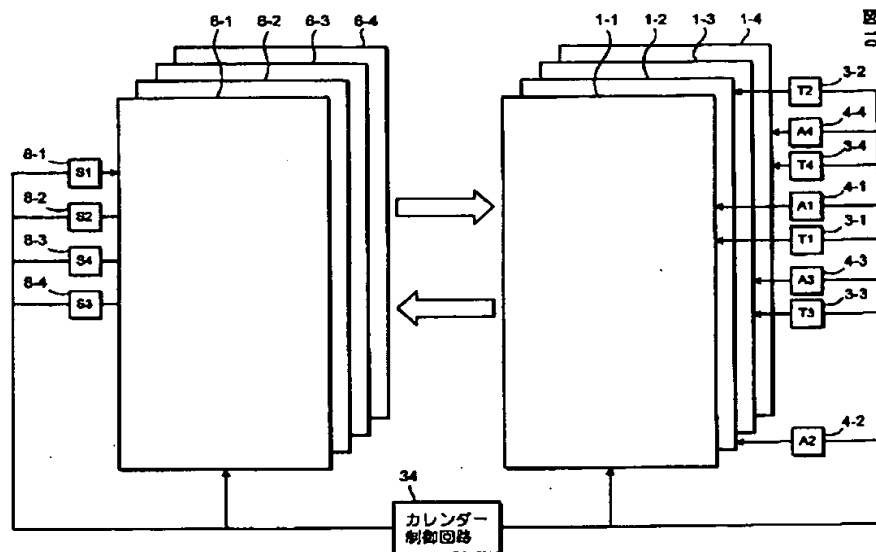


【図 8】

図 8

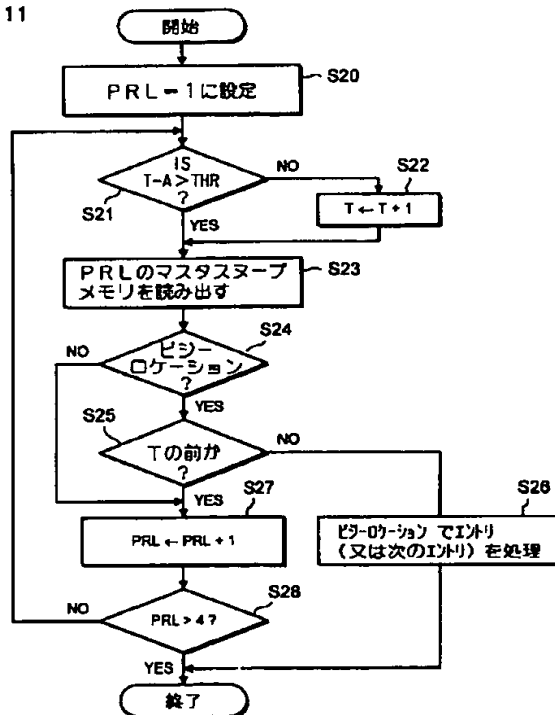


【図10】

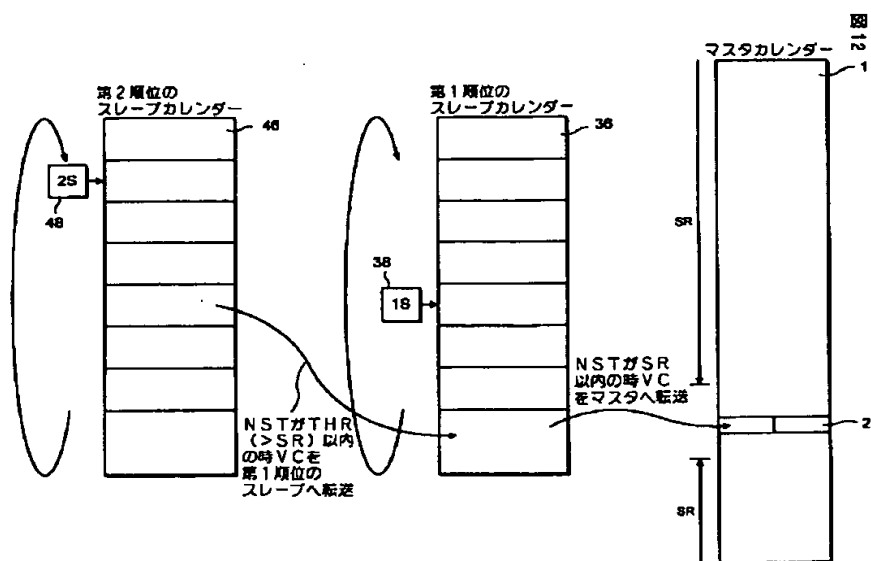


【図11】

図 11



【図12】



フロントページの続き

(72) 発明者 サイモン テイモシー スミス
イギリス国, マンチェスター エム21 0
ティーキュー, コールトン, リズバーン
アベニュー 1